A Finite-Automaton Based Stream Cipher As a Quasigroup Based Cipher*

Pál Dömösi
Faculty of Informatics, University of Debrecen
H-4028 Debrecen, Kassai út 26, Hungary
Institute of Mathematics and Informatics, University of Nyíregyháza
H-4400 Nyíregyháza, Sóstói út 36, Hungary
domosi@unideb.hu

Adama Diene
Department of Mathematical Science, United Arab Emirates University,
P.O. Box 15551, Al Ain, Abu Dhabi, United Arab Emirates
diene@uaeu.ac.ae

In this paper we show that a recently published finite automaton stream cipher can be considered as a quasigroup based stream cipher. Some additional properties of the discussed cipher are also given.

1 Introduction

In this paper we consider the finite automaton based stream cipher published by Dömösi and Horváth [2], and we show in details that this cipher can be considered as a stream cipher based on quasigroup. Some additional properties are also discussed. The stream cipher in [2] works, in short, as follows. The cipher consists of a cryptographically secure pseudorandom generator and a finite automaton without outputs having the same input and state sets. During the encryption the plaintext is read in sequentially character by character. After getting the next (initially the first) plaintext character, the system gets simultaneously the next (initially the first) pseudorandom string of a fixed length which is also an input string of the key-automaton. The corresponding ciphertext character will coincide with the state of the key-automaton into which this pseudorandom input string takes the automaton from the state which coincides with the corresponding plaintext character. The decryption works similarly, using a so-called inverse key-automaton instead of the key automaton such that the input strings will be the mirror images of the corresponding pseudorandom strings.

We note that there are several variants of the quasi group based ciphers. Fortunately, we did not find such a solution in the literature that is equivalent to the solution we are discussing. (Detailed overviews and summaries of quasi group based ciphers can be found, for example, in [1, 6, 7].)

2 Preliminaries

We start with some standard concepts and notation. For all notions and notation not defined here we refer to the monographs [3, 4, 5, 7] and the reviews [1, 6]. By an alphabet we mean a finite nonempty set. The elements of an alphabet are called letters. A word over an alphabet Σ is a finite string consisting of letters of Σ. The *length* of a word w, in symbols $|w|$, means the number of letters in w when each letter is

*This work was supported by United Arab Emirates Program for Advanced Research (UAEU UPAR) Grant No. G00003431.

H. Bordihn, G. Horváth, G. Vaszil (Eds.): 12th International Workshop on Non-Classical Models of Automata and Applications (NCMA 2022)
EPTCS 367, 2022, pp. 81–87, doi:10.4204/EPTCS.367.6

counted as many times as it occurs. The string consisting of zero letters is called the *empty word,* written by λ. By definition, $|\lambda| = 0$. The mirror image w^R of the word $w = a_1 \cdots a_n, a_1 \ldots, a_n \in \Sigma$ is the word $w^R = a_n \cdots a_1$. By definition, $\lambda^R = \lambda$. Furthermore, for every nonempty word w, denote by $\overrightarrow{w}$ the last letter of w. ($\overrightarrow{\lambda}$ is not defined.) The set of all nonempty words over an alphabet Σ will be denoted by Σ^+. In addition, we put $\Sigma^* = \Sigma^+ \cup \{\lambda\}$. By an automaton we mean a finite deterministic automaton without outputs. In other words, by an automaton we mean a system $\mathbf{A} = (A, \Sigma, \delta)$ with a finite set A of states, a finite set Σ of inputs, and the transition function $\delta : A \times \Sigma \to A$. We assume that the transition function of $\mathbf{A}$ is given in the form of transition table, where the lines of this table are denoted by the elements of the input set Σ and the columns of this table are denoted by the elements of the state set A. Therefore, for every input x and state a, $\delta(a,x)$ is at the intersection of the row denoted by x and the column denoted by a.

3 Automata and Quasigroups

Given an automaton $\mathbf{A} = (A, \Sigma, \delta)$, let $a \in A, a_1, \ldots, a_n \in \Sigma$ and suppose $b_1 = \delta(a, a_1), b_2 = \delta(b_1, a_2), \ldots,$ $b_n = \delta(b_{n-1}, a_n)$. Then we shall use the notation $\delta(a, a_1 \cdots a_n) = b_1 \cdots b_n$. (Thus we may use the notation $\overrightarrow{\delta(a, a_1 \cdots a_n)}$ for the above considered b_n.) Moreover, by definition, $\delta(a, \lambda) = \lambda$. In what follows we consider automata having the same state and input sets, i.e., we assume $A = \Sigma$. A groupoid $Q = (A, *)$ is a structure consisting of the nonempty set A and the binary operation $*$ over A. Therefore, the concept of automaton $\mathbf{A} = (A, A, \delta)$ coincides with the concept of groupoid $Q = (A, *)$ having $a * b = \delta(b, a)$ for every pair $a, b \in A$. A groupoid $Q = (A, *)$ is called a quasigroup if for every pair $a, b \in A$ there exists unique $x, y \in A$ such that $a * x = b$ and $y * a = b$. It is easy to see that quasigroups satisfy both of the cancellation properties, i.e., for every triplet $a, b, c \in A$, $a * b = a * c$ implies $b = c$ (left cancellation), and $a, b, c \in A$, $a * b = c * b$ implies $a = c$ (right cancellation). It is said that $\backslash$ is the left inverse operation on Q if for every triplet $a, b, c \in A$, $a * b = c$ if and only if $b = a \backslash c$. Analogously, $/$ is the right inverse operation on Q if for every triplet $a, b, c \in A$, $b * a = c$ if and only if $b = c/a$. Then the groupoid $Q_{LI} = (A, \backslash)$ is the left inverse quasigroup of Q, and similarly, the groupoid $Q_{RI} = (A, /)$ is the right inverse quasigroup of Q. [1]

4 Latin squares and key automata

A Latin square of order n is an $n \times n$ matrix (with n rows and n columns) in which the elements of an n-state set $\{a_0, a_1, \ldots, a_{n-1}\}$ are entered so that each element occurs exactly once in each fixed row, and each fixed column, respectively. We say that $\mathbf{A} = (A, \Sigma, \delta)$ is a key automaton if for every pair of distinct states a, b and pair of distinct inputs x, y, $\delta(a,x)$ differs from $\delta(b,x)$ and $\delta(a,x)$ also differs from $\delta(a,y)$. Obviously, in this case the transition table of a key automaton forms a Latin square and there is a one-to-one correspondence between the key automata $\mathbf{A} = (A, A, \delta)$ and quasigroups $Q = (A, *)$ having the property $\delta(a,b) = b * a$ for every pair $a, b \in A$ of elements in A and vice versa. Given a key automaton $\mathbf{A} = (A, A, \delta)$, let us define the automaton $\mathbf{B} = (A, A, \delta^{-1})$ such that for every pair $a, b \in A, \delta^{-1}(\delta(a,b), b) = a$. Then we say that $\mathbf{B}$ is the inverse key automaton of the key automaton $\mathbf{A}$.

***Proposition* 1** Every key automaton has exactly one inverse key automaton.
Proof. Consider a key automaton $\mathbf{A} = (A, A, \delta_{\mathbf{A}})$. Suppose that $\mathbf{B} = (A, A, \delta_{\mathbf{B}})$ and $\mathbf{C} = (A, A, \delta_{\mathbf{C}})$ are inverse key automata of $\mathbf{A}$ such that $\mathbf{B} \neq \mathbf{C}$. Then there are $a, b \in A$ having $\delta_{\mathbf{B}}(a,b) \neq \delta_{\mathbf{C}}(a,b)$. Put

[1] It is easy to show that both of Q_{LI}, Q_{RI} are quasigroups

$x = \delta_{\mathbf{B}}(a,b)$ and $y = \delta_{\mathbf{C}}(a,b)$. Then we have $a = \delta_{\mathbf{A}}(x,b) = \delta_{\mathbf{A}}(y,b)$ contradicting the assumption that **A** is a key automaton. This completes the proof.
QED.

***Proposition* 2** Every inverse key automaton is also a key automaton.
Proof. Consider a key automaton $\mathbf{A} = (A,A,\delta)$ and its inverse key automaton $\mathbf{B} = (A,A,\delta^{-1})$. First we suppose that there are states a,b,c with $\delta^{-1}(a,b) = \delta^{-1}(a,c)$ and $b \neq c$. Put $d = \delta^{-1}(a,b) = \delta^{-1}(a,c)$.By our assumptions, this implies $\delta(d,b) = \delta(d,c) = a$ with $b \neq c$ contradicting the assumption that **A** is a key automaton. Thus $b \neq c$ implies $\delta^{-1}(a,b) \neq \delta^{-1}(a,c)$ for every $a \in A$. Next we suppose that there are states a,b,c with $\delta^{-1}(a,b) = \delta^{-1}(c,b)$ and $a \neq c$. Put $d = \delta^{-1}(a,b) = \delta^{-1}(c,b)$.By our assumptions, this implies $\delta(d,b) = a = c$ contradicting to $b \neq c$. Thus $a \neq c$ implies $\delta^{-1}(a,b) \neq \delta^{-1}(c,b)$ for every $b \in A$. Therefore, we received that **B** is also a key automaton.
QED.

By the definition of inverse key automaton and Poposition 2 we have as follows.

***Corollary* 3** Let **B** be the inverse key automaton of the key automton **A**. Then **A** is the inverse key automaton of **B**.

***Proposition* 4** Given a key automaton $\mathbf{A} = (A,A,\delta)$, its inverse key automaton $\mathbf{A^{-1}} = (A,A,\delta^{-1})$, a state $a \in A$, and a string $a_1 \cdots a_n, a_1 \ldots, a_n \in A$, we have $\delta(a, a_1 \cdots a_n) = b_1 \cdots b_n$ if and only if $\delta^{-1}(b_n, a_n \cdots a_1)$ $= b_{n-1} \cdots b_1 a$.
Proof. Let $\mathbf{A} = (A,A,\delta)$ be an arbitrary finite automaton and consider (nonempty and finite) strings $a_1 \cdots a_n$ and $b_1 \cdots b_n$ consisting of the elements $a_1, \ldots, a_n, b_1, \ldots, b_n$ of A. In addition, $\delta(a,a_1) = b_1$ if and only if $\delta^{-1}(b_1,a_1) = a$, where δ^{-1} denotes the transition function of the inverse key automaton $\mathbf{A^{-1}}$ of **A**. Similarly, $\delta(b_1,a_2) = b_2$ if and only if $\delta^{-1}(b_2,a_2) = b_1$. Thus we obtain that $\delta(a,a_1a_2) = b_1b_2$ if and only if $\delta^{-1}(b_2,a_2a_1) = b_1a$. Repeating this procedure we get our statement.
QED.

We have the following consequence of this statement.

***Proposition* 5** Given a key automaton $\mathbf{A} = (A,A,\delta)$, its inverse key automaton $\mathbf{A^{-1}} = (A,A,\delta^{-1})$, a state $a \in A$, and a string $a_1 \cdots a_n, a_1 \ldots, a_n \in A$, we have $\overrightarrow{\delta(a,a_1 \cdots a_n)} = b_n$ if and only if $\overrightarrow{\delta^{-1}(b_n, a_n \cdots a_1)} = a$.

5 Quasigroups

We shall use the following statement.

***Proposition* 6** Given a quasigroup $Q = (A,*)$, its left inverse quasigroup $Q_{LI} = (A, \setminus)$, moreover, $a_1, \ldots,$ $a_n, a, b \in A$. Then $a_n * (\cdots * (a_2 * (a_1 * a)) \cdots) = b$ if and only if $a_1 \setminus (\cdots \setminus (a_{n-1} \setminus (a_n \setminus b)) \cdots) = a$.
Proof. We will prove our statement by induction. Suppose $n = 1$. Then, by definition, $b = a_1 * a$ if and only if $a = a_1 \setminus b$. Thus, it is enough to show than if our statement holds for any given case $n = m$, then it must also hold for the next case $n = m+1$. Thus, assume that for every $b, c, a_2, \ldots, a_{m+1} \in A$, $a_{m+1} * (\cdots * (a_2 * c) \cdots) = b$ if and only if $a_2 \setminus (\cdots \setminus (a_{m+1} \setminus b)) \cdots) = c$. Set $c = a_1 * a$ for some $a_1 \in A$. Then $a_{m+1} * (\cdots * (a_2 * (a_1 * a)) \cdots) = b$ if and only if $a_2 \setminus (\cdots \setminus (a_{m+1} * b)) \cdots) = a_1 * a$.

Substituting $a_2 \setminus (\cdots \setminus (a_{m+1} * b)) \cdots)$ for b, then we receive $b = a_1 * a$ which follows $a = a_1 \setminus b$ by definition. This implies $a_1 \setminus a_2 \setminus (\cdots \setminus (a_{m+1} * b)) \cdots) = b$ as we stated.
QED.

The following statement is obvious.

***Proposition* 7** Given a quasigroup $Q = (A, *)$, let $Q_{LI} = (A, \backslash)$ be its left inverse quasigroup. Then for every pair $x, y \in A$, $x \backslash (x * y) = y, x * (x \backslash y) = y$. Given a key automaton $\mathbf{A} = (A, A, \delta)$, the corresponding quasigroup $Q = (A, *)$ ordered to $\mathscr{A}$ is defined by $a * b = \delta(a, b), a, b \in A$.

***Theorem* 8** Let $Q = (A, *)$ be the corresponding quasigroup ordered to the key automaton $\mathbf{A} = (A, A, \delta)$. Then the left inverse quasigroup of Q is the corresponding quasigroup ordered to the inverse key automaton of $\mathbf{A} = (A, A, \delta)$ and vice versa.
Proof. Consider a key automaton $\mathbf{A} = (A, A, \delta)$ and its inverse key automaton $\mathbf{B} = (A, A, \delta^{-1})$.

Then the corresponding quasigroup $Q = (A, *)$ ordered to $\mathbf{A}$ has the property $\delta(a, b) = b * a$ for every pair $a, b \in A$. Similarly, the corresponding quasigroup $R = (A, \bigodot)$ ordered to $\mathbf{B}$ has the property $\delta^{-1}(c, d) = d \bigodot c$ for every pair $c, d \in A$.

By definition, for every pair $a, b \in A$, $\delta^{-1}(\delta(a, b), b) = a$. This implies $b \bigodot (b * a) = a$. Then $c = b * a$ implies $b \bigodot c = a$.

Next we assume $b \bigodot (b * a) = a$ and $c \neq b * a$ with $b \bigodot c = a$. Put $d = b * a$. Then we get $b \bigodot d = a$ with $b \bigodot c = a$ and $d \neq c$. In other words, $\delta^{-1}(d, b) = \delta^{-1}(c, b)(= a)$ with $d \neq c$. But then, by definition, the inverse key automaton $\mathbf{B}$ is not a key automaton. This statement contradicts to Proposition 2.
QED.

***Proposition* 9** Given a quasigroup $Q = (A, *)$ ordered to the key automaton $\mathbf{A} = (A, A, \delta)$,let $a, a_1, \ldots, a_n \in A$. Then $\overrightarrow{\delta(a, a_1 \cdots a_n)} = b$ for some $b \in A$ if and only if $a_n * (a_{n-1} * (\cdots * (a_1 * a) \cdots)) = b$.
Proof. By our conditions, we have in order, $\overrightarrow{\delta(a, a_1)} = \delta(a, a_1) = a_1 * a$, $\overrightarrow{\delta(a, a_1 a_2)} = \overrightarrow{\delta(a, a_1)\delta(\delta(a, a_1), a_2)} = \delta(\delta(a, a_1), a_2) = a_2 * (a_1 * a)$, and inductively, $\overrightarrow{\delta(a, a_1 \cdots a_n)} = a_n * (a_{n-1} * (\cdots * (a_1 * a) \cdots))$. Using these observations, by definition, $b_1 = \delta(a, a_1)$ if and only if $b_1 = a_1 * a$. Similarly, $b_2 = \delta(b_1, a_2)$ if and only if $b_2 = a_2 * (a_1 * a)$. Repeating this procedure, we have $b_n = \delta(b_{n-1}, a_n)$ if and only if $b_n = a_n * (a_{n-1} * (\cdots * (a_1 * a) \cdots))$.

Let $b = b_n$. Then we get as we stated.
QED.

6 A finite automaton based stream cipher

Consider a pseudorandom number generator, a key automaton $\mathbf{A} = (A, A, \delta)$, and its inverse key automaton $\mathbf{A^{-1}} = (A, A, \delta^{-1})$. The main idea of the discussed cipher is the following.

6.1 Encryption

Let $p_1 \cdots p_n, p_1, \ldots, p_n \in A$ be a plaintext and let $r_1, \ldots, r_n \in A^+$ be pseudorandom strings of the same fixed length $m \geq 1$ generated by a given pseudorandom number generator starting by a seed r_0. We note that $|r_0|, \ldots, |r_k| = m$ holds for a fixed positive integer m.

The ciphertext will be $c_1 \cdots c_n, c_1, \ldots, c_n \in A$ with $c_1 = \overrightarrow{\delta(p_1, r_1)}, \ldots, c_n = \overrightarrow{\delta(p_n, r_n)}$.

6.2 Decryption

Let $c_1 \cdots c_n, c_1, \ldots, c_n \in A$ be a ciphertext and let $r_1, \ldots, r_n \in \Sigma^+$ be the same pseudorandom strings generated by the pseudorandom number generator starting by a seed r_0.

The decrypted plaintext will be $p_1 \cdots p_n$ with $p_1 = \overrightarrow{\delta^{-1}(c_1, (r_1)^R)}, \ldots, p_n = \overrightarrow{\delta^{-1}(c_n, (r_n)^R)}$.

The next statement shows the correctness of the discussed finite automaton-based encryption and decryption procedure.

***Theorem* 10** Let $p_1 \cdots p_n, p_1, \ldots, p_n \in A$ be a plaintext and let $r_1, \ldots, r_n \in A^+$ be pseudorandom strings of the same fixed length $m \geq 1$ generated by a given pseudorandom number generator starting by a seed r_0. Moreover, let $\mathbf{A} = (A, A, \delta)$ be a key automaton and let $\mathbf{A^{-1}} = (A, A, \delta^{-1})$ be its inverse key automaton. If $c_1 \cdots c_n$ is the ciphertext generated by the above finite automaton encryption procedure then $p_1 \cdots p_n$ is the only plaintext which can be generated by the above finite automaton based decryption procedure (assuming that the pseudorandom generator of the cipher generates the same sequence $r_1, \ldots, r_n$ of the pseudorandom strings during the encryption and also during the decryption).

Proof. Consider a key automaton $\mathbf{A} = (A, A, \delta)$, its inverse key automaton $\mathbf{A^{-1}} = (A, A, \delta^{-1})$, a state $a \in A$, and a string $r \in A$. By Proposition 5 we have $\overrightarrow{\delta(a, r)} = b_n$ if and only if $\overrightarrow{\delta^{-1}(b_n, r^R)} = a$.[2]

By our construction, for every $i = 1, \ldots, n, c_i = \overrightarrow{\delta(p_i, r_i)}$. By Proposition $5 p_i = \overrightarrow{\delta^{-1}(c_i, r_i^R)}$. In sum, $c_1 \cdots c_n = \overrightarrow{\delta(p_1, r_1)} \cdots \overrightarrow{\delta(p_n, r_n)}$ which, by Proposition 5 is possible if and only if $p_1 \cdots p_= \overrightarrow{\delta^{-1}(c_1, r_1^R)} \cdots \overrightarrow{\delta^{-1}(c_n, r_n^R)}$. This completes the proof.

QED.

7 A quasigroup based stream cipher

Consider again a cryptographically secure pseudorandom number generator, moreover a quasigroup $Q = (A, *)$ and its left-inverse $Q_{LI} = (A, \backslash)$. The main idea of the discussed cipher is the following.

7.1 Encryption

Let m be a fixed positive integer, and in order to have $P_T = p_1 p_2 p_3 \cdots p_n,\ p_1, \ldots, p_n \in A$, $K = k_{1,1} k_{1,2} \cdots k_{1,m} \cdots k_{n,1} k_{n,2} \cdots k_{n,m}$, $C_T = c_1 c_2 \cdots c_n$ as the plaintext P_T to be encrypted, a pseudorandom sequence K is generated by the cryptographically secure pseudorandom number generator as the keystream to be used for encryption, and the resulting ciphertext C_T respectively. Then a way of encrypting P_T with the keystream K to obtain the corresponding C_T is as follows:

$c_1 = k_{1,m} * (\cdots * (k_{1,2} * (k_{1,1} * p_1)) \ldots), c_2 = k_{2,m} * (\cdots * (k_{2,2} * (k_{2,1} * p_2)) \ldots), \ldots c_n = k_{n,m} * (\cdots * (k_{n,2} * (k_{n,1} * p_n)) \ldots)$.

7.2 Decryption

Let m be the same fixed positive integer again as in Subsection 6.1, and in order to have the same $C_T = c_1 c_2 \cdots c_n$, $K = k_{1,1} k_{1,2} \cdots k_{1,m} \cdots k_{n,1} k_{n,2} \cdots k_{n,m}$ as in Section 6.1, as the ciphertext to be decrypted, a pseudorandom sequence K is generated by the cryptographically secure pseudorandom number generator as the keystream to be used for encryption,and the resulting plaintext $P_T = p_1 p_2 p_3 \cdots p_n$, respectively. Then a way of decrypting C_T with the keystream K to obtain the corresponding P_T back is as follows: $p_1 = k_{1,1} \backslash (\cdots \backslash (k_{1,m-1} \backslash (k_{1,m} \backslash c_1)) \ldots), p_2 = k_{2,1} \backslash (\cdots \backslash (k_{2,m-1} \backslash (k_{2,m} \backslash c_2)) \ldots), p_n = k_{n,1} \backslash (\cdots \backslash (k_{n,m-1} \backslash k_{n,m} \backslash c_n)) \ldots)$, where $\backslash$ denotes the quasigroup operation and $\backslash$ denotes the corresponding left inverse quasigroup operation.

[2]Recall that for every $r \in A^+, r^R$ denotes the mirror image of r.

Next we show that the work of the discussed stream cipher can be written easily by using automata-theoretic disciplines like in [2]. In more details, the next statement shows the correctness of the discussed finite qusigroup based encryption and decryption procedure.

***Theorem* 11** Let $p_1 \cdots p_n,\ p_1, \ldots, p_n \in A$ be a plaintext and let $r_1, \ldots, r_n \in A^+$ be random strings of the same fixed length $m \geq 1$ generated by a cryptographically secure pseudorandom number generator starting by a seed r_0. Moreover, let $Q = (A, *)$ be a quasigroup and let $Q_{LI} = (A, \backslash)$ be its left inverse quasigroup. If $c_1 \cdots c_n$ is the ciphertext generated by the above qusigroup based encryption procedure then $p_1 \cdots p_n$ is the only plaintext which can be generated by the above qusigroup based decription procedure (assuming that the pseudorandom generator of the cipher generates the same sequence $r_1, \ldots, r_n$ of the pseudorandom strings during the encryption and also during the decryption).

Proof. By Proposition 6, for every quasigroup $Q = (A, *)$, its left inverse quasigroup $Q_{LI} = (A, \backslash)$, and $a_1, \ldots, a_n, a, b \in A$ it holds that $a_n * (\cdots * (a_2 * (a_1 * a)) \cdots) = b$ if and only if $a_1 \backslash (\cdots \backslash (a_{n-1} \backslash (a_n \backslash b)) \cdots) = a$. Let a denote the i^{th} character p_i of the plaintext, moreover, let b denote the i^{th} character c_i of the ciphertext for some $i \in \{1, \ldots, n\}$. In addition, let $a_1 \cdots a_m$ denote the i^{th} pseudorandom string r_i generated by the pseudorandom generator of the cipher. Then, by Proposition 6, we have that for every $i = 1, \ldots, n$, that p_i is the only i^{th} plaintext character which can be generated by the discussed qusigroup based decryption procedure whenever c_i is the i^{th} ciphertext character which can be generated by the discussed qusigroup based encryption procedure and r_i is the same i^{th} pseudorandom string generated by the pseudorandom generaton in both of the encryption and the decryption. Therefore, if $c_1 \cdots c_n$ is the ciphertext generated by the considered qusigroup based encryption procedure then $p_1 \cdots p_n$ is the only plaintext which can be generated by the considered qusigroup based decription procedure (assuming that the pseudorandom generator of the cipher generates the same sequence $r_1, \ldots, r_n$ of the pseudorandom strings during the encryption and also during the decryption).This completes the proof. ***QED.***

8 Quasigroups in Cryptography

The most of the quasigroup-based cryptosystems essentially work based on the following principle [1, 6].

Given a quasigroup $Q = (A, *)$, its left inverse quasigroup $Q_{LI} = (A, \backslash)$, let $\ell \in A$ be a fixed element, which is called a leader. (Actually, $\ell \in A$ can be considered as the secret seed of the encryption/decryption).

Encryption. Let $p_1 \cdots p_n$ be a plaintext of $n \geq 1$ letters $p_1, \ldots, p_n \in A$. Compute $c_1 = \ell * p_1, c_2 = c_1 * p_2, \ldots, c_n = c_{n-1} * p_n$. Then the ciphertext is $c_1 \cdots c_n$.

Decryption Let $c_1 \cdots c_n$ be a plaintext of n letters $c_1, \ldots, c_n \in A$. Compute $p_1 = \ell \backslash c_1, p_2 = c_1 \backslash c_2, \ldots, p_n = c_{n-1} \backslash c_n$. Then the recovered plaintext is $p_1 \cdots p_n$.

Cryptanalyses of this classical quasigroup-based cipher was made by M. Vojvoda [8]. He showed that this cipher is not resistant to chosen plaintext attack and ciphertext-only attack in contrast to our discussed solution. There are several known variants of this classical quasigroup cipher applying special quasigroups, and/or multiple leaders, multi-round ciphering, etc. [1].

9 Conclusion

This paper shows that the cipher in [2] can be considered as a quasigroup-based stream cipher. By this observation, we can easily compare it with the other quasigroup-based ciphers. It can be concluded that our solution is mainly different from them.

In order, to achieve a higher speed of encryption/description operation, m should be as small as possible. Therefore, next we should analyse this cipher with $m = 1$. Thus, using the finite automaton-based form, we should consider again a $c_1 = \delta(p_1, k_1), \ldots, c_n = \delta(p_n, k_n)$, and the description can be given by $p_1 = \delta^{-1}(c_1, k_1), p_2 = \delta^{-1}(c_2, k_2), \ldots, p_n = \delta^{-1}(c_n, k_n)$, where δ denotes the transition function of the key automaton and δ^{-1} denotes the transition function of the inverse key automaton.

The equivalent quasigroup-based form of this cipher can also be considered as follows: $c_1 = p_1 * k_1, c_2 = p_2 * k_2, \ldots, c_n = p_n * k_n$, and the description can be given by $p_1 = c_1 \setminus k_1$, $p_2 = c_2 \setminus k_2, \ldots, p_n = c_n \setminus k_n$, where $*$ denotes the quasigroup operation and $\setminus$ denotes the corresponding right inverse quasigroup operation.

A further challenge of research is to show the security of the proposed cipher using several theoretical and experimental investigations regarding the length of the applied pseudorandom sequences, the number of rounds in multi-round encryption and decryption, and some other parameters.

References

[1] Chauhan, D., Gupta, I., and Verma, R. Quasigroups and their Application in Cryptography. Cryptologia, 2020, 1–39, doi:10.1080/01611194.2020.1721615.

[2] Dömösi, P. and Horváth, G. A Novel Stream Cipher Based on Deterministic Finite Automaton. Freund, R., Mráz, F., Prusa, D. (eds.): Proc. Ninth Workshop on Non-Classical Models of Automata and Applications (NCMA 2017), Short Papers Wien, Ausztria : Technical University of Vienna, 2017, 11–16.

[3] Dömösi, P. and Nehaniv, C. L. Algebraic theory of automata networks. An introduction. SIAM Monographs on Discrete Mathematics and Applications, 11. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005, doi:10.1137/1.9780898718492.

[4] Hopcroft, J.E., Motwani, R., and Ullman, J. D. Introduction to Automata Theory (second edition). Addison-Wesley Series in Computer Science, Addison-Wesley Co., Reading, MA, 2001, doi:10.1145/568438.568455.

[5] Menezes, A. J., Oorschot, P. C., Vanstone, S. A. Handbook of Applied Cryptography. CRC Press Series on Discrete Mathematics and Its Applications, CRC Press LLC, Boca Raton, FL, USA, 1996, 2001, 2008, doi:10.1201/9780429466335.

[6] Shcherbacov, V. A. Quasigroups in cryptology.. Comput. Sci. J of Moldova, vol. 17 no. 2(50), 2009.

[7] Shcherbacov, V. A. Elements of Quasigroup Theory and Applications. Chapman & Hall/CRC Press,London, New York, ISBN 978-1-4987-2155-4, 2017, doi:10.1201/9781315120058.

[8] Vojvoda, M. Stream ciphers and hash functions - analysis of some new design approaches. PhD thesis, Slovak University of Technology, July, 2004.